FACTA UNIVERSITATIS
Series: **Electronics and Energetics** Vol. 35, N° 4, December 2022, pp. 603-617
https://doi.org/10.2298/FUEE2204603D

Original scientific paper

CHAOTIC SEISMIC SIGNAL MODELING BASED ON NOISE AND EARTHQUAKE ANOMALY DETECTION

Leila Dehbozorgi, Reza Akbari-Hasanjani, Reza Sabbaghi-Nadooshan

Department of Electrical Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran

Abstract. *Since ancient times, people have tried to predict earthquakes using simple perceptions such as animal behavior. The prediction of the time and strength of an earthquake is of primary concern. In this study chaotic signal modeling is used based on noise and detecting anomalies before an earthquake using artificial neural networks (ANNs). Artificial neural networks are efficient tools for solving complex problems such as prediction and identification. In this study, the effective features of chaotic signal model is obtained considering noise and detection of anomalies five minutes before an earthquake occurrence. Neuro-fuzzy classifier and MLP neural network approaches showed acceptable accuracy of 84.6491% and 82.8947%, respectively. Results demonstrate that the proposed method is an effective seismic signal model based on noise and anomaly detection before an earthquake*.

Key words: *Artificial neural networks, chaos, earthquake, entropy, prediction, seismic signal processing, wavelet transforms*

1. INTRODUCTION

Earthquake prediction is a branch of seismology and should be distinguished from earthquake warning systems which provide a real-time warning to regions that might be affected. The purpose of a chaotic signal model considering noise and detection of anomalies before an earthquake is to warn of an impending major earthquake to reduce death and destruction. In the 1970s, scientists were optimistic that a practical method for predicting earthquakes would soon be found [1]. However, further devastating earthquakes occurred that caused destruction and loss of life exceeding 6,300 persons in the M7.2 1995 Kobe earthquake in Japan, 15,000 in the M7.4 1999 Izmit earthquake in Turkey, and over 30,000 in the M6.7 2003 Bam earthquake in Iran [2].

There are many common methods of detecting anomalies before an earthquake which use artificial neural networks (ANNs), genetic programming (GP), and radial basis function networks. Artificial neural networks have applications in areas such as identification, prediction, and image processing. In Ref. [3], the back propagation neural network and new mark displacement analysis examined the earthquake risk in the Manjil-rudbar damaged area

Received April 4, 2022; revised July 26, 2022; accepted August 10, 2022
Corresponding author: Reza Sabbaghi-Nadooshan
Department of Electrical Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran
E-mail: r_sabbaghi@iauctb.ac.ir

in 1990. In order to evaluate earthquake signals, it is better to use factual information than the null hypothesis. Ref. [4] considered a model for noisy signal and detecting anomalies before an earthquake using ANNs and got acceptable results on Ghir station in Iran.

Researchers have developed software for short-term earthquake prediction using pressure reduction and temperature rise, which has resulted in 70.5% accurate forecasting in Japan. The accuracy of this network is not optimal for predicting [5]. In Ref. [6] used location related parameters in the neural network to predict earthquakes in Iran. The researchers in Ref. [7] used the deep learning model of DLEP for earthquake prediction, which used explicit and implicit features. There is no suitable time frame for earthquake prediction. In Ref. [8] the neural network is discussed to predict the arrival time of P-wave earthquake occurrence in Taiwan.The time frame for earthquake prediction is concise. In Ref. [9] has used a GRNN neural network to predict earthquakes on the Iranian plateau. In Ref. [22] examines the possibility of using the DLIS algorithm to identify and reconstruct the location, size, and thickness distribution of several complex defects. In Ref. [23], it has used 8 mini-stations of the new region located in North Sumatra which it uses the SVM model (one of the machine learning tools in digital signal processing) to distinguish seismic activities. However, the proposed model has acceptable accuracy but the amount of data to be tested can be increased and the used more data is necessary for test network performance.

Ref. [24] used the Deep Learning to predict earthquakes and P-wave has been investigated, but the time frame for earthquake prediction is concise.

Ref. [25] also used the Deep Learning and Neural Network to predict earthquakes. The period of 3 seconds before the earthquake is intended to predict the earthquake that is a concise time to forecast.

Ref. [26] suggested the augmented linear mixing model (ALMM) method. The most of the focus of this article is on image processing, object recognition, and classification. In this purpose, a dictionary is defined to model spectral variables. This paper focuses more on image processing, not signal processing. The dictionary has little similarity with the training data in neural network. But, the application of methods used for image processing needs more investigation in signal processing.

Ref. [27] proposed a model called Fourier-based rotation-invariant feature boosting (FRIFB) to increase the speed of calculations and reduce complexity. In this way, the Fourier is calculated in polar coordinates and then the subsequent analyses are performed. In this article, we have defined several extracted frequency features, which are used almost the same way but with some differences as in the above article. For example, to extract newer and different features, we applied signal divider features to FFT and PSD, and then extracted statistical features for each part. It is explained in more detail in sections 2-3.

In Ref. [10], two feature groups were compared to detect anomalies before an earthquake. Those consisted of 54 and 87 features. The accuracy values for data classifier and MLP neural network are equal to 60.6383% and 55.8511% for the feature matrix with dimensions of 54 and 87, with a total of 626 records. This method employed much more data than previous methods. It does not have the desired accuracy, but there is a time frame before the earthquake to predict it.

Most previous articles use concise time frame to predict earthquakes, and the number of features is minimal and related to geological features. For this reason, it is impossible to make an accurate decision about the types of effective features in the occurrence of an earthquake.

This study aims to determine the most desirable characteristic matrix for detecting anomalies within 5 minutes before an earthquake and chaotic signal modeling.

In this study, a new class of effective features is developed for chaotic signal modeling before an earthquake using intelligent networks with a more extensive database for generalization than previous methods. Then a model is evaluated for noisy signal and detection of anomalies before an earthquake using neuro-fuzzy and MLP classifiers. The innovation of this article is the use of more data for training and testing the classifiers, considering 5 minutes before an earthquake to predict it, using different features than previous articles and comparing the performance of two neuro-fuzzy networks and MLP classifiers. Most papers use only geological or frequency-related features. In this article, we tried to examine the different types of features and determine the effectiveness or ineffectiveness of each.

The rest of the paper is organized as follows: Section 2 discusses the basic concepts of the features and ANN structure. Section 3 proposes the design method and discusses the simulation results. Section 4 concludes with the obtained result.

2. DATA AND METHODOLOGY

Chaotic signal modeling based on noise was employed with neuro-fuzzy and MLP classifiers and using a large amount of data and some new features. The seismic waves are processed to detect anomalies before an earthquake onset. The method is divided into six main stages (in Fig. 1): (1) Considering the earthquake onset to select an observation window and detect anomalies; (2) Slice the rest of the signal into two sections; (3) High pass filtering of the signals to reject baseline drift; (4) Feature extraction from the filtered signal; (5) Feed the feature vector to the intelligent networks; (6) After training and testing the classifiers, select the effective features using UTA algorithm [11,12].

The selected signals were processed using a high pass Butterworth filter to remove baseline drift in the signals and the cut-off frequency (FC) was set at 0.04 Hz [4]. There was not enough evidence showing how an earthquake related to a known feature, so a mixture of time, time-scale, and chaos features were extracted, and the effective features were selected after achieving acceptable accuracy [4, 10]. The whole process of the algorithm is shows in Fig. 1.

2.1. Features

2.1.1. Statistical Features

The ten statistical features evaluated are mode, mean, variance, covariance, maximum data, minimum data, signal standard deviation, median, deviation of string factor from symmetry (SK) and stretch factor. The ‘SK’ and ‘k’ features represented as follow. Where x_i and $\bar{x}$ denote the signal and the mean of signal, respectively, and N is the number of data [13].

$$sk = \left[(\sum(\chi_i - \bar{\chi})^3)/\mathrm{N}\right] \Big/ \left[\sqrt{((\sum(\chi_i - \bar{\chi})^2)/\mathrm{N})}\right]^3 \tag{1}$$

$$k = E = \left[(\sum(\chi_i - \bar{\chi})^4)/\mathrm{N}\right] \Big/ \left[\sqrt{((\sum(\chi_i - \bar{\chi})^2)/\mathrm{N})}\right]^4 - 3 \tag{2}$$

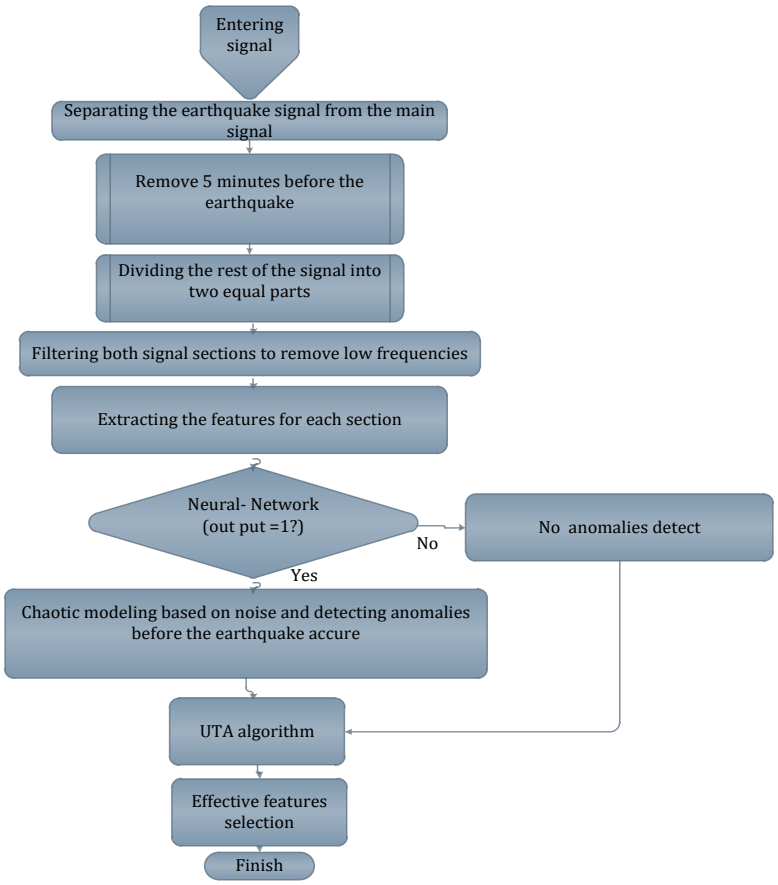

Fig. 1 Flowchart of chaotic modeling based on noise and detecting anomalies before the earthquake

2.1.2. Chaos Features

Chaotic systems are highly dependent on initial conditions. In other words, if two trajectories start very close to each other, they diverge from each other rapidly and exponentially if and only if their processes have chaotic behavior. The difference between the two trajectories after the time period of T is measured as the Lyapunov exponent (λ). Where that X_0 is a point on a trajectory at time t and $X_0 + \Delta x_0$ is the point near to X_0 on a different trajectory where Δx_0 approaches zero and presents the initial amount of separation between the two points.

$$\lambda = \ell im_{\tau\to\infty}(1/\tau)\cdot \ell n(\left|\Delta X(X_0,\tau)\right|/\left|\Delta X_0\right|) \tag{3}$$

There are three states for the Lyapunov exponent (λ):
(1) λ>0: the system is chaotic.
(2) λ<0: the system is not chaotic.
(3) λ =0: the system reaches steady state condition [13].

2.1.3. Signal Divider

A signal divider applied to classify of data between the maximum and minimum signal values. The signal divided into 16 equal classes and the amount of available data in each class is extracted as the feature.

2.1.4. Entropy

Entropy is a measure of the system disorder. Entropy H(X) of discrete random variable X is evaluated as following [14], so that P(x) is the probability of X occurrence.

$$H(X) = -\sum_x P(X) \cdot \ell og_2(P(X)) \tag{4}$$

2.1.5. Discrete Wavelet Transform (DWT)

Wavelet transform can be seen as the projection of a signal into a set of basic functions named wavelets. A wavelet transform includes a function based on the mother wavelet function and has an excellent localization characteristic in the time-scale domain [15]. Most of the energy in a wavelet function is concentrated in a short interval and is damped quickly. Of the various types of wavelet functions, the Daubechies wavelet transform is one of the most common. In wavelet transforms, the signal passes through an internal filter and is divided into a low-frequency (CA) and a high-frequency (CD) component. The DWT of signal x[n] is defined based on approximation coefficient w_φ [j_0,k] and detail coefficient w_ψ [j,k], as it is shown as follows. Where n=0,1,2,…, M-1, k=0,2,…,2j-1 and j=0,1,2,…,J-1, and M is the number of samples to be transformed using wavelet function.

$$W_\Phi[j_0,k] = (1/\sqrt{M}) \cdot \sum_n x[n] \cdot \Phi_{j_0,k} \tag{5}$$

$$W_\Psi[j,k] = (1/\sqrt{M}) \cdot \sum_n x[n] \cdot \Psi_{j_0,k}\ , \quad \text{for } j \geq j_0 \tag{6}$$

The basic functions $\varphi_{j,k}$ [n], and $\psi_{i,k}$ [n] are defined as follow. where φ[n] is the scaling function and ψ[n] is the wavelet function [4,16].

$$\Phi_{j,k}[n] = 2^{j/2} \cdot \Phi[2^j \cdot n - k] \tag{7}$$

$$\Psi_{j,k}[n] = 2^{j/2} \cdot \Psi[2^j \cdot n - k] \tag{8}$$

The Daubechies 2 wavelet transform is implemented in the next five steps. The output of each array is selected using half of the inputs selected at each step. The statistical values are used as features in each step.

2.1.6. Fast Fourier Transform (FFT)

Using the equation 9, Fast Fourier Transform (FFT) for an n×n matrix is calculated [17]. The statistical features and data classifier for the FFT of the signal evaluated as features.

$$X(k) = \sum_{n=0}^{N-1} X(n) \cdot (\exp(-j2\pi k / N))\ \ k=0, 1, 2, \ldots, N \tag{9}$$

2.1.7. Power Spectral Density (PSD)

The power spectral density (PSD) function shows the strength of variation (energy) as a function of frequency. It shows at which frequencies variations are strong and at which

frequencies variations are weak. The energy is obtained within a specific frequency range by integrating PSD in the frequency range. The computation of PSD is done directly by computing autocorrelation function R(τ) and then transforming it. The results are demonstrated in the following formulas for signal s(t).

$$P(t) = s(t)^2 \tag{10}$$

$$S(f) = \int_{-\infty}^{+\infty} R(\tau) \cdot (\exp(-j2\pi\tau)) \cdot d\tau = F(R(\tau)) \tag{11}$$

The power of the signal in a frequency band can be calculated as:

$$P = \int_{f1}^{f2} S(f) \cdot df + \int_{-f1}^{-f2} S(f) \cdot df \tag{12}$$

Afterward, statistical features of the signal's PSD were derived as PSD features.

2.1.8. Trajectory

A trajectory is a path followed by an object moving through space as a function of time. In this present study, a signal with N pieces of data is presumed. Each part of the signal is depicted as $\{x(t_1), x(t_2), \ldots, x(t_N)\}$ such that $t_1, t_2, \ldots, t_N$ refers to the data stored in a time series [18]. First, the x(n+1) to x(n) graph is represented as a signal trajectory and then is divided into 16 houses. The number of pieces of data stored in each house in a matrix is a feature.

2.2. Classification Networks

2.2.1. Multilayer Perceptron (MLP) Network

Multilayer Perceptron (MLP) is a well-known feed-forward neural network that is used for classification usually because of its good performance. Generally, an MLP contains input and output layers and one or more hidden layers. After forming the structure of a network, the neurons are connected by linking weights and they are trained using a training algorithm (Fig. 2) [19].

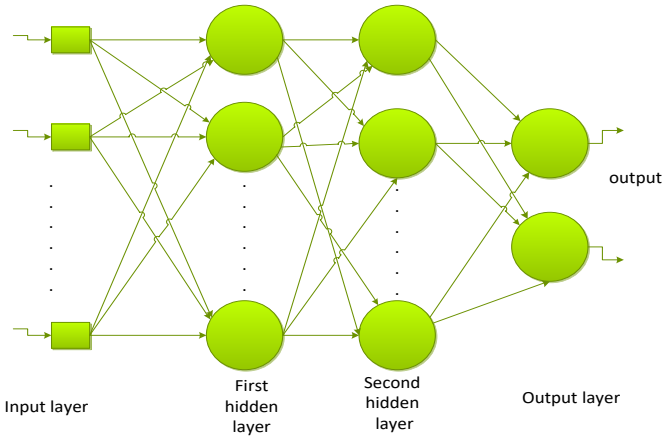

Fig. 2 The structure of multi-layer perceptron network

2.2.2. Neuro-Fuzzy Classification Networks

Fuzzy systems use two significant paradigms: fuzzy logic and neural networks [20]. Fuzzy logic programming in MATLAB software includes conditional statements. The neural network consists of several nodes which are connected to each other by weights (Fig. 3).

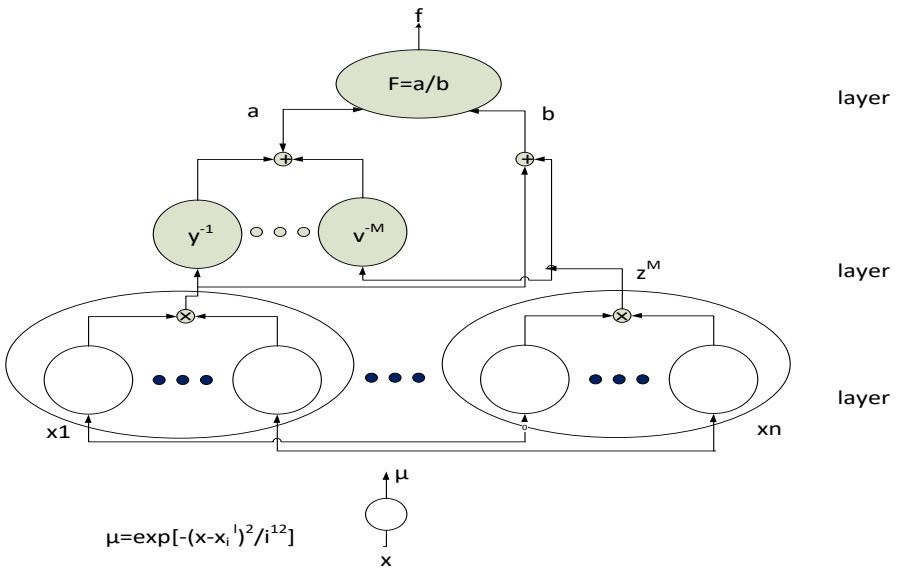

Fig. 3 Network representation of the fuzzy system [10]

$$f(x)=\frac{\left(\sum_{l=1}^{M} y^{-l}\cdot\left[\prod_{i=1}^{N}\exp(-((x_i-x_i^{-1})/\sigma_i^{l}))\right]\right)}{\left(\sum_{l=1}^{M}\left[\prod_{i=1}^{N}\exp(-((x_i-x_i^{-1})/\sigma_i^{l})^2)\right]\right)} \tag{13}$$

Here, three parameters σ_i^l, $y^{-1}$and x_i^{-1} define in the phase of learning and must be determined to design a neuro-fuzzy system, and M is the number of rules considered. Input x passes through a product Gaussian operator to become z^l, then result of this stage passes through summation operator *b* and weighted operator a. Finally, output *f* is calculated [20].

$$Z^l=\prod_{i=1}^{N}\exp(-((x_{0i}^{p}-x_i^{-l})/\sigma_i^{l})) \tag{14}$$

$$b=\sum_{i=1}^{M} z^l \tag{15}$$

$$a=\sum_{l=1}^{M} y^{-l}\cdot(q)\cdot Z^l \tag{16}$$

$$f=a/b \tag{17}$$

2.3. Feature Selection

In the UTA algorithm, the average of one feature in all instances is calculated. Then the selected feature in all input vectors is replaced by the calculated mean value. Then trained network is tested with the new features and new matrix. If the system cognition is

decreased, that feature is effective, but if the result doesn't change or improve, that feature is considered ineffective (noisy feature) and should be removed from the input vector [11].

3. Analysis of Results

The database contains 760 records at 5 to 7 on the Richter scale from the International Institute of Earthquake Engineering and Seismology for 21 earthquake recording stations in Iran (between 2004 and 2010). The sampling frequency is 50 Hz (Fig. 4). Table 1 shows the date, time, geographical location, depth, and magnitude of each earthquake.

Table 1 Characteristics of 5 to 7 Richter Earthquakes Recorded Between 2004 and 2009

Date of occurrence			Time of occurrence			Magnitude and geographical characteristics			
Year	Month	Day	Hour	Minute	Second	Latitude	Longitude	Depth	Magnitude
2004	10	6	11	14	26.1	28.8	57.9	14.1	5.2
2004	10	7	12	54	56.1	28.4	57.2	15.9	5
2004	10	7	21	46	15.2	37.3	54.5	16.8	6.2
2004	10	16	10	4	33.9	33.5	45.7	18	5
2005	3	13	3	31	27.3	27.3	61.5	54.8	6.1
2005	5	1	18	58	38.8	30.8	56.9	14.2	5.1
2005	5	14	18	4	57.1	30.7	56.6	14.1	5.2
2005	6	19	4	46	4.5	33.1	58.2	15	5.2
2005	8	9	5	9	19.7	28.8	52.6	18	5
2005	11	27	16	30	39.1	27.0	55.7	14.1	5.2
2005	11	29	5	57	3	37.5	54.6	15	5
2005	12	26	23	15	51.1	32.1	49.1	32.9	5.2
2005	12	27	21	53	15	28.1	56.1	15	5.1
2006	2	18	11	3	31.5	30.7	55.8	14.1	5
2006	2	28	7	31	3.4	28.1	56.7	18	5.8
2006	3	25	7	28	57.3	27.5	55.8	15.8	5.5
2006	3	25	9	55	16	27.6	56.0	15.9	5.1
2006	3	25	10	0	37	27.4	55.7	15	5
2006	3	30	19	36	18	33.6	48.9	15	5.1
2006	3	31	1	17	2.3	33.6	48.9	14.1	6.1
2006	3	31	11	54	2.6	33.8	48.7	17.5	5.2
2006	6	28	21	2	9.2	26.8	55.9	10	5.6
2006	7	18	23	27	5.5	26.2	61.1	46	5
2006	11	5	20	6	40.2	37.4	48.8	14.1	5
2007	3	26	6	36	50	29.1	58.4	14.1	5
2007	6	18	14	29	49.4	34.5	50.8	17.3	5.6
2008	3	9	3	51	6.4	33.3	59.1	17.9	5
2008	8	27	21	52	39.9	32.3	47.3	32.5	5.6
2008	9	10	11	0	35.1	26.9	55.7	6.7	5.8
2008	10	25	20	17	16.9	26.6	54.8	14.2	5.1
2008	12	7	13	36	20.8	26.9	55.7	11	5.2
2008	12	9	15	9	27.4	27.0	55.8	15	5
2009	7	22	3	53	2.6	26.7	55.8	14.2	5.2
2009	10	4	21	50	49.6	31.8	49.4	15	5.1

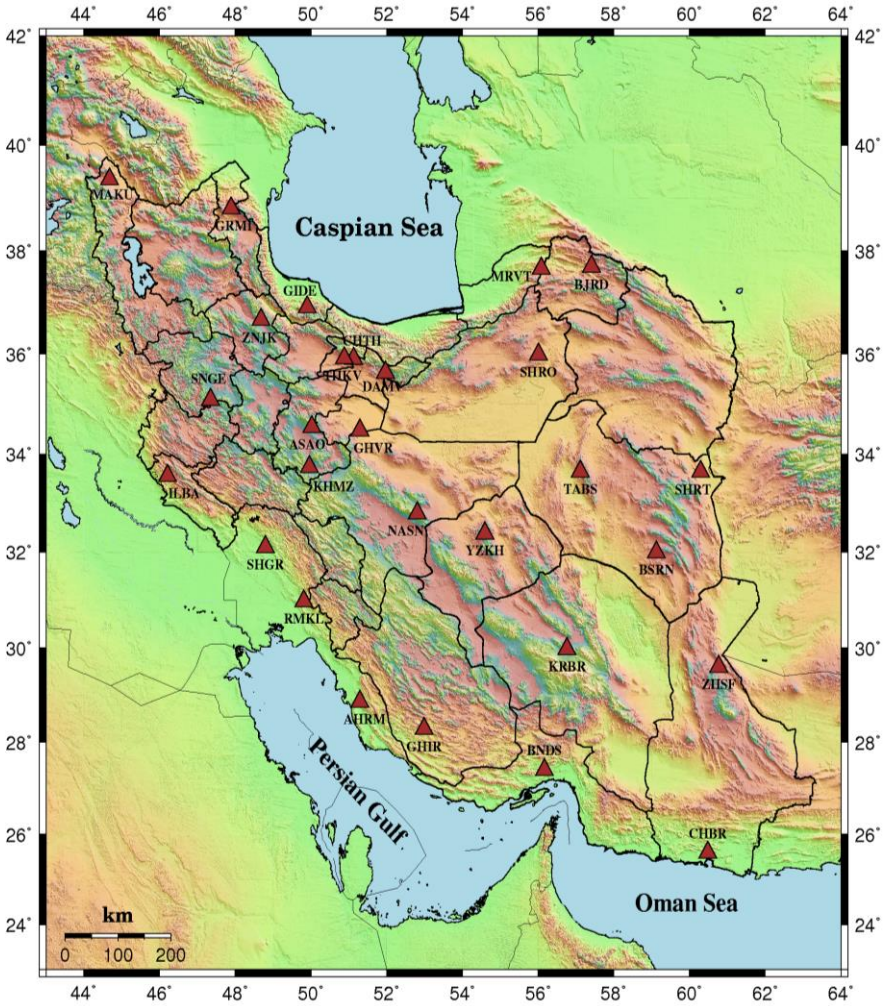

Fig. 4 Distribution of stations in the IRAN Broadband National Center Seismology [21]

For training, 70% of the data were randomly selected and the remaining 30% of the data was used for testing in MATLAB R2017a software. Initially, for anomaly detection, 380 records that have 20 minutes of the signal and have the property that an earthquake has happened after were selected as the sig1 group and 380 records equal in length by the sig1 group were selected which no earthquake has occurred in the next following five minutes after them and five minutes before the earthquake in each record deleted (Fig. 5). The first five minutes of sig2 and last five minutes of sig1 separated to extract features for the feature vector. A fourth-order high pass Butterworth filter was applied to remove the low frequency (FC = 0.04) and then the signals normalized (Fig. 6).

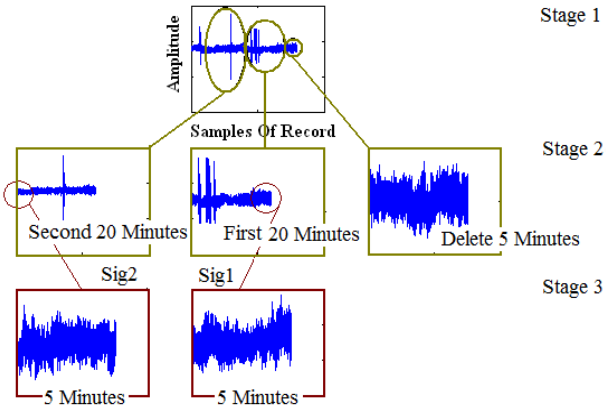

Fig. 5 Classification of seismic signals before the earthquake, the earthquake does not happen after Sig2 and happen 5 minutes after Sig1

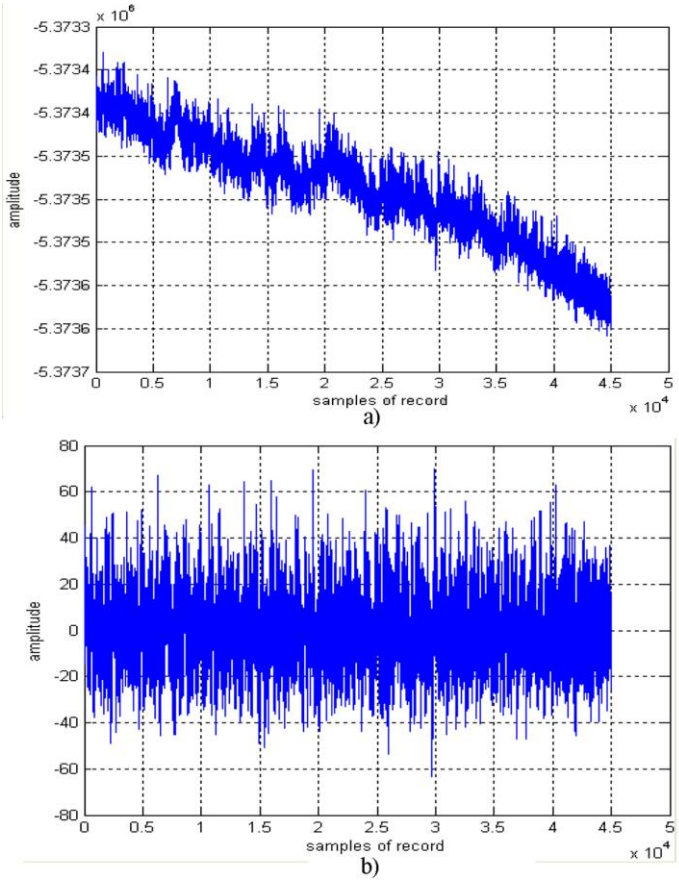

Fig. 6 a) The original signal, b) the filtered signal

After filtering the 15000 samples, the statistical features are derived for each record using the CHAOSTEST.m in MATLAB [4]. CHAOSTEST.m tests the positive existence of the dominant Lyapunov exponent λ and local Lyapunov exponents. The second output parameter is H, which is the result comparing λ and α. The value of p is the observing probability. Another output is Orders, which gives the triplet (L, m, q), minimizes the Schwarz information criterion to obtain the best coefficients and calculates λ. The confidence interval (CI) for λ is determined at level α (α is a fixed number with a default value of 0.05).

The DWT is implemented for five steps and eight statistical features (mode, mean, variance, covariance, maximum, minimum, median, and signal standard deviation) is saved for each step. The signal divider is evaluated for FFT and eight statistical features are calculated for FFT and PSD, respectively. Moreover, the x(n+1) to x(n) graph is provided as a signal trajectory and then divided into 16 houses. The amount of data stored in each place is saved as a feature.

Input feature vector has 260 values per instance. The neuro-fuzzy classifier has 260 inputs, 14 neurons (rules), and one output. The threshold of the classification in neuro-fuzzy classifier is 0.49. Furthermore, the MLP neural network has 260 neurons in the input layer, two hidden layers, and an output layer consisting of two neurons. Neuro-fuzzy classifier and MLP neural network were successfully trained in MATLAB and the testing results are presented for both networks. The networks have one output; each output value uniquely represents one category (0: no earthquake; 1: earthquake). After training, both classifiers were tested and then 3503 iterations of training, the results indicated that the neuro-fuzzy classifier was better than the MLP network and could detect anomalies five minutes before an earthquake with an acceptable accuracy of 84.6491% (Fig. 7; Table 2).

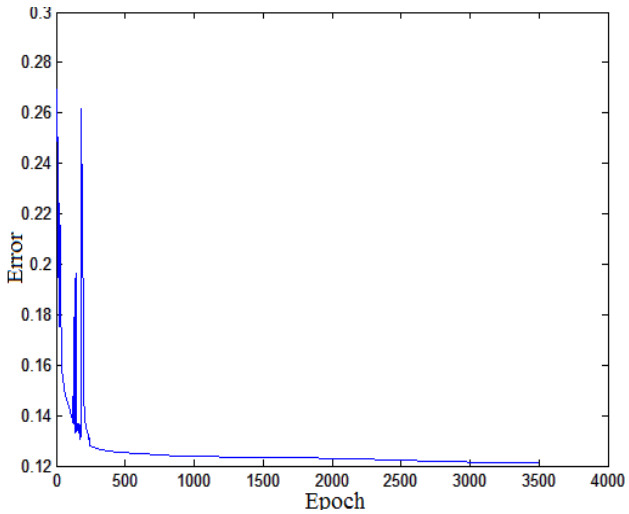

Fig. 7 Difference between the output of Neuro-Fuzzy classifier and real output after 3503epoch

Table 2 Neuro-fuzzy classifier's performance compare with MLP before feature selection

Classifier	Neuro-Fuzzy Classifier	Multilayer Perceptron (MLP) network
Accuracy	84.6491%	81.1404%
Sensitivity	71.93%	80.70%
Specificity	97.37%	81.58%
Average Error	0.1237	0.1691 0.1663

Fig. 8 compares the neuro-fuzzy classifier and MLP performance before feature selection. This figure shows that the neuro-fuzzy classifier is produced better results for accuracy.

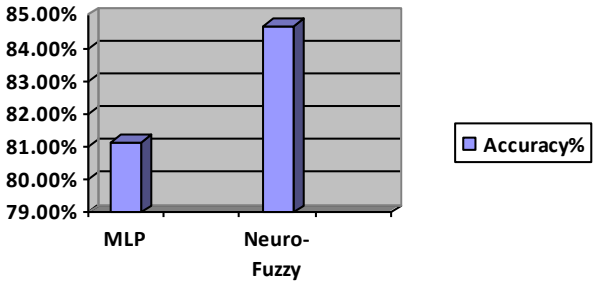

Fig. 8 Neuro-fuzzy classifier's performance compared to MLP before feature selection

After training and testing, the UTA algorithm implemented for feature selection and ineffective features were deleted. This algorithm decreased the input vector dimensions to 150 for the MLP network and 29 for the neuro-fuzzy classifier. Both classifiers were trained and tested again (Table 3). Table 4 shows some of the more effective features of both classifiers. Results show that frequency characteristics are priorities for both classifiers and the neuro-fuzzy classifier produced better results for accuracy and sensitivity.

Table 3 Neuro-fuzzy classifier's performance compare with MLP after feature selection

Classifier	Neuro-Fuzzy Classifier	Multilayer Perceptron (MLP) network
Accuracy	84.6491%	82.8947%
Sensitivity	74.56%	71.05%
Specificity	94.74%	94.74%
Average Error	0.1512	0.1782 0.1580

Table 4 Some of more effective features after implementation of UTA algorithm for neuro-fuzzy classifier and mlp neural network

MLP Neural Network	Neuro-Fuzzy Classifier
Mean of angle (FFT)	Mean of angle (FFT)
Mean of angle of normalize (FFT)	Median of Entropy
Max of data	Covariance of CA (DWT)
Mean of abs (FFT)	Signal standard deviation of (PSD)
Max of CA (DWT)	Mean of CA (DWT)
Mean of CD (DWT)	Trajectory

Fig. 9 shows the results of the accuracy of this study compared with other studies. Amount of accuracy optimization compared to previous articles is obtained using the following formula:

$$\text{Improvement (\%)} = \left(1 - \frac{\text{present implementation result}}{\text{previousimplementationresult}}\right) \times 100 \quad (18)$$

Accuracy:

Improvement (%) (this study, [4]) = (1–(84.6491/60.8491))×100 = 39.079%
Improvement (%) (this study, [10]) = (1–(84.6491/60.6383)×100 = 39.59%
Improvement (%) (this study, [7]) = (1–(84.6491/50)×100 = 69.2982%
Improvement (%) (this study, [8]) = (1–(84.6491/75)×100 = 12.8654%

It can be seen that the accuracy in the proposed design is more optimal than the previous articles. The amount of improvement is even close to 70% (Fig. 10).

Fig. 11 shows the results of the present study improved for the neuro-fuzzy classifier after the implementation of the new features in this study. It shows that the neuro-fuzzy classifier has performs better than the MLP network.

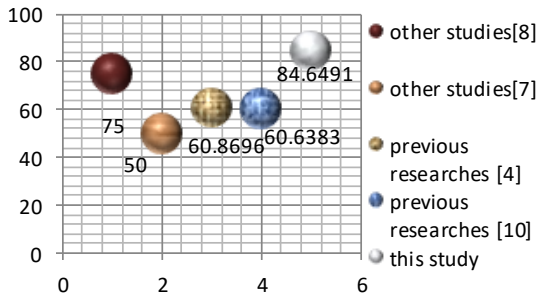

Fig. 9 Neuro-fuzzy classifier's performance for this study (TS) compared with the other studies

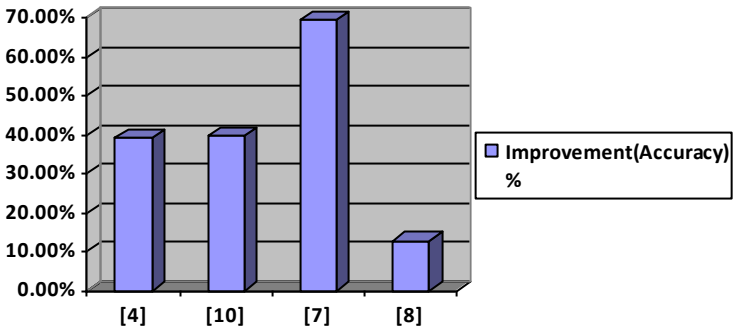

Fig. 10 Comparison of the accuracy improvement of the proposed design with previous articles

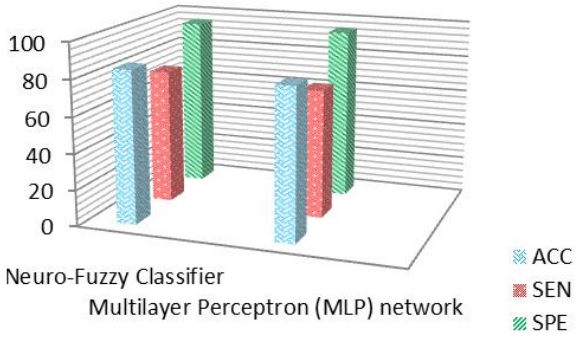

Fig. 11 Neuro-Fuzzy classifier's performance compared with MLP after feature selection

4. CONCLUSION

In this article, the proposed method can detect anomalies before an earthquake by using new features. One of the innovations of this article is extracting new features. Also, we considered a longer period of time than the rest of the articles to detect the anomaly before the earthquake then evaluated two types of classifiers. Finally, we chose the best network and the most optimal features.

The proposed method provided a new matrix of features that was capable of chaotic signal modeling based on noise and detection of anomalies during the five minutes before the earthquake with an acceptable accuracy of 84.6491%. Moreover, the results indicate that the UTA algorithm decreased input feature dimensions without loss of accuracy. The selected features demonstrated that chaotic signal modeling based on noise and detecting anomalies before an earthquake is very dependent on frequency features, followed by entropy, trajectory, chaotic and statistical features. Future work would be to collect more earthquake data globally, add more frequency-dependent parameters to the feature vector, and use committee machines to increase the classification accuracy. It is also possible to extract a new feature from combination of two or three features for example, the combination of entropy and classification and frequency features or other possible combinations.

REFERENCES

[1] R. J. Geller, D. D. Jackson, Y. Y. Kagan and F. Mulargia, "Earthquakes cannot be predicted," *Science*, vol. 275, pp. 1616-1617, 1997.

[2] S. Uyeda, T. Nagao, and M. Kamogawa, "Short-term earthquake prediction: Current status of seismo-electromagnetics", *Tectonophysics*, vol. 470, no. 3-4, pp. 205-213, 2009.

[3] A. M. Rajabi, M. Khodaparast, and M. Mohammadi, "Earthquake-induced landslide prediction using back-propagation type artificial neural network: case study in northern Iran", *Natural Hazards*, vol. 110, no. 1, pp. 679-694, 2022.

[4] L. Dehbozorgi, "Case Study of Seismic Signals for Ghir Station before the Earthquake", *Bulletin of Earthquake Science and Engineering*, vol. 5, no. 4, pp. 131-143, 2019.

[5] H. Shiraishi, "Developing and Validating Earthquake Prediction Software", *International Journal of Engineering and Techniques*, vol. 8, pp. 63-69, 2022.

[6] M. Yousefzadeh, S. A. Hosseini, and M. Farnaghi, "Spatiotemporally explicit earthquake prediction using deep neural network", *Soil Dynamics and Earthquake Engineering*, vol. 144, p. 106663, 2021.

[7] R. Li, X. Lu, S. Li, H. Yang, J. Qiu, and L. Zhang, "DLEP: a deep learning model for earthquake prediction," In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), 2020, pp. 1-8.
[8] Y.-J. Chiang, T.-L. Chin, and D.-Y. Chen, "Neural Network-Based Strong Motion Prediction for On-Site Earthquake Early Warning", *Sensors*, vol. 22, no. 3, pp. 704, 2022.
[9] S. Yaghmaei-Sabegh, "Earthquake ground-motion duration estimation using general regression neural network", *Scientia Iranica*, vol. 25, no. 5, pp. 2425-2439, 2018.
[10] L. Dehbozorgi and F. Farokhi, "Notice of retraction: Effective feature selection for short-term earthquake prediction using neuro-fuzzy classifier", In Proceedings of the 2010 Second IITA International Conference on Geoscience and Remote Sensing, 2010, vol. 2, pp. 165-169.
[11] J. Utans, J. Moody, S. Rehfuss, and H. Siegelmann, "Input variable selection for neural networks: Application to predicting the US business cycle", In Proceedings of 1995 Conference on Computational Intelligence for Financial Engineering (CIFEr), 1995, pp. 118-122.
[12] M. F. Redondo and C. H. Espinosa, "A comparison among feature selection methods based on trained networks", In Proceedings of the Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No. 98TH8468), 1999, pp. 205-214.
[13] K. Majumdar and M. H. Myers, "Amplitude suppression and chaos control in epileptic EEG signals", *Computational and Mathematical Methods in Medicine*, vol. 7, no. 1, pp. 53-66, 2006.
[14] S. Byun *et al.*, "Entropy analysis of heart rate variability and its application to recognize major depressive disorder: A pilot study", *Technology and Health Care*, vol. 27, no. S1, pp. 407-424, 2019.
[15] K. Sui and H.-G. Kim, "Research on application of multimedia image processing technology based on wavelet transform", *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, pp. 1-9, 2019.
[16] A. Qin, Z. Shang, J. Tian, Y. Wang, T. Zhang, and Y. Y. Tang, "Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification", *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 2, pp. 241-245, 2018.
[17] A. L. Zheleznyakova, "Physically-based method for real-time modelling of ship motion in irregular waves", *Ocean Engineering*, vol. 195, pp. 106686, 2020.
[18] Y. Xue, P. J. Ludovice, and M. A. Grover, "Dynamic coarse graining in complex system simulation", In Proceedings of the 2011 American control conference, 2011, pp. 5031-5036.
[19] N. Singh and R. Khan, "Speaker Recognition and Fast Fourier Transform", *International Journal*, vol. 5, no. 7, 2015.
[20] R. Tabbussum and A. Q. Dar, "Performance evaluation of artificial intelligence paradigms—artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference system for flood prediction", *Environmental Science and Pollution Research*, vol. 28, no. 20, pp. 25265-25282, 2021.
[21] International Institute of Earthquake Engineering and Seismology. [Online]. Available: http://www.iiees.ac.ir.
[22] J. Tong, M. Lin, X. Wang, J. Li, J. Ren, L. Liang, Y. Liu, "Deep learning inversion with supervision: A rapid and cascaded imaging technique", *Ultrasonics*, 122, 106686, 2022.
[23] M. Sinambelaa, M. Situmoranga, K.Tarigana, S. Humaidia, Makmur Siraitb, "Waveforms Classification of Northern Sumatera Earthquakes for New Mini Region Stations Using Support Vector Machine", *Advanced science Engineering information Technology*, vol.11, no. 2, 2021.
[24] W. Yanwei, L. Xiaojun, W. Zifa, et al., "Deep learning for P-wave arrival picking in earthquake early warning", *Earthq. Eng. Eng.*, vol. 20, pp. 391-402, 2021.
[25] M. S. Abdalzaher, M. S. Soliman, S. M. El-Hady, A. Benslimane, et al., "A Deep Learning Model for Earthquake Parameters Observation in IoT System-Based Earthquake Early Warning", *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8412-8424, 2022.
[26] D. Hong, N. Yokoya, J. Chanussot, X. X. Zhu, "An Augmented Linear Mixing Model to Address Spectral Variability for Hyperspectral Unmixing", *IEEE Transactions on Image Processing*, *vol.* 28, no. 4, 2019.
[27] X. Wu, D. Hong, J. Chanussot, Y. Xu, R. Tao, Y. Wang, "Fourier-Based Rotation-Invariant Feature Boosting: An Efficient Framework for Geospatial Object Detection", *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 2, 2020.